DOCUMENT RESUME

ED 142 990 CS 003 635

AUTHOR Yeager, Robert F.
TITLE The Reading Machine.
PUB DATE May 77
NOTE 32p.; Paper presented at the Annual Meeting of the International Reading Association (22nd, Miami Beach, Florida, May 2-6, 1977)

EDRS PRICE MF-$0.83 HC-$2.06 Plus Postage.
DESCRIPTORS *Beginning Reading; *Computer Assisted Instruction; Grade 1; Primary Education; *Reading Instruction; *Teaching Machines; *Teaching Techniques
IDENTIFIERS *PLATO Elementary Reading Curriculum Project

ABSTRACT

This paper describes lessons in beginning reading, developed by the PLATO Elementary Reading Curriculum Project (PERC), for use with first graders. The lessons were developed to reflect a number of specific principles, including: (1) all responses should be meaningful; (2) remedial feedback should be kept to a minimum; (3) students must always be forced to make the correct response; and (4) students should be able to pace their own instruction. Specific lessons from the PERC series are used to illustrate these principles and to demonstrate the unique uses of a computer for teaching reading. These uses turn the computer into a "reading machine," letting the first graders feel that they are in control of their computer terminal, rather than controlled by it. (AA)

****
* Documents acquired by ERIC include many informal unpublished *
* materials not available from other sources. ERIC makes every effort *
* to obtain the best copy available. Nevertheless, items of marginal *
* reproducibility are often encountered and this affects the quality *
* of the microfiche and hardcopy reproductions ERIC makes available *
* via the ERIC Document Reproduction Service (EDRS). EDRS is not *
* responsible for the quality of the original document. Reproductions *
* supplied by EDRS are the best that can be made from the original. *
****

ED142990

U S DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

THE READING MACHINE

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Robert F. Yeager

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC) AND THE ERIC SYSTEM CONTRACTORS."

Robert F. Yeager
252 Engineering Research Laboratory
University of Illinois
Urbana, Illinois 61801

Paper presented at the Annual Meeting of the International Reading Association, Miami, May 5, 1977.

THE READING MACHINE

Robert F. Yeager

In his book, The Use and Misuse of Computers in Education, Allan Ellis calls the computer an "unfinished machine". It is a tool in search of a purpose; in order to become useful, something has to be put inside the computer which can give it a purpose. If physics equations are put into the computer, it becomes a "physics machine"; bookkeeping algorithms change a computer into a "bookkeeping machine".

This paper will explain how to turn a computer into a "reading machine", that is, a machine which can be used to teach children how to read. It will describe some of the principles of lesson design developed by the PLATO Elementary Reading Curriculum project (PERC) at the University of Illinois; PERC has produced between twenty and twenty-five hours of beginning reading instruction which is currently in use in fifteen primary grade classrooms in Champaign and Urbana.

PERC lessons are different than other reading lessons developed for use with computer-assisted instruction (CAI). For one thing, PERC lessons were designed for first graders; there is no other large body of CAI materials for such young students. It is difficult to write lessons for young children; they lack the experience to muddle through lessons which are not designed as well as they should be. But by the same token, PERC has been forced to attend to the details of lesson design; and from this comes PERC's conception of the appropriate and unique role of CAI in beginning reading instruction.

Paper presented at the Annual Meeting of the International Reading Association, Miami, May 5, 1977.

Another thing which sets PERC lessons apart is that they use the PLATO technology. Both the PLATO hardware and software are highly advanced; they challenge the lesson designer's imagination rather than limit it.

But the most important thing which differentiates PERC lessons from other CAI lessons is that PERC lessons have been purposely designed from a particular perspective. That perspective has little to do with reading per se, but it does affect how a child learns to read. This paper will describe that perspective; give some examples of PERC lessons which exemplify it; and discuss the implications of that approach for teaching beginning reading with a computer.

The Nature of CAI

The most important thing to remember about a computer is that it is a machine; and machines are meant to be used by people. That means that people should control the machines, not vice versa. And for PERC, it means that first graders should feel that they are in control of their computer terminals, not vice versa.

The first lesson which students usually get on PLATO is Freddy Frog (figure 1). Freddy hops across the screen. Then the audio tells the students that they can make Freddy hop just by touching him with their finger. Students typically cannot quite believe that, but after a little coaxing they find out that THEY really can make Freddy Frog jump just by touching him.

Then Freddy (via audio) tells the students that they can tell him where to go! He hops wherever they touch on the screen. From this very first lesson, PERC students learn that they are in control of the computer terminal.

Figure 1. Freddy Frog "hopping" across the screen.

One of the students favorite lessons is the "Elevator" (figures 2 and 3). The students can move an elevator up and down by touching the appropriate "buttons" on the screen. Little figures call for the elevator at different floors; the students have to press the right button to make the elevator move to waiting figure; then, when the person gets on the elevator, the students are told (via audio) to go up or down.

The "Elevator" is a good example of how students can acquire skills (the sightwords "up" and "down", and their meanings) while they are exercising control over their terminal.

The "Animal Maker" lesson (figure 4) allows students to create new and exotic animals while they practice their sightwords. They can put two words together, such as "alligator and hippo"; then the computer shows them what an "alligator- hippo" would look like.

Not only does the "Animal Maker" practice sight words, but it also prepares the students for the notion of compound words. But most important, it shows the students the power of words. Imagine being able to cross a dog and a cat just by typing "dogcat"!

Branching stories are another example of a unique use of the computer in teaching reading. In "The Race" (figures 5 through 10), a dog and a cat get into an argument about which one is the fastest; so they decide to have a race. The students decide how fast each one will run, and whether a bridge along the race route will be up or down. Then the dog and cat race and the story unfolds just as the students said it should.

All PERC lessons typically allow students to repeat lessons; so once students have seen how a branching story works out one way, they can go back and try it a different way. They use the computer as a tool; and they gain experience and learn the importance of comprehending text.

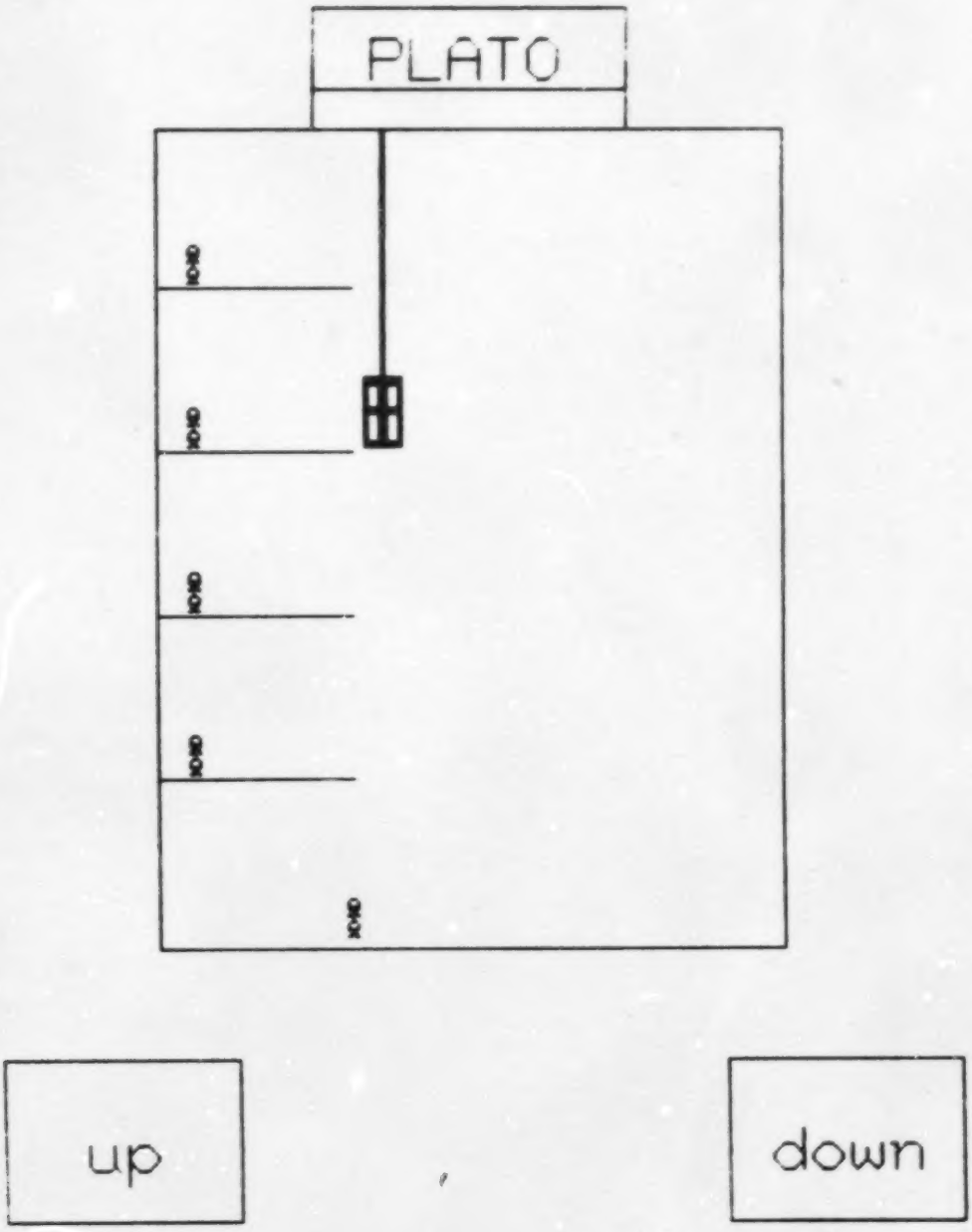


Figure 2. The figure on the bottom floor says, "Send the elevator to me, please." The student has to read the words and understand which button to push.

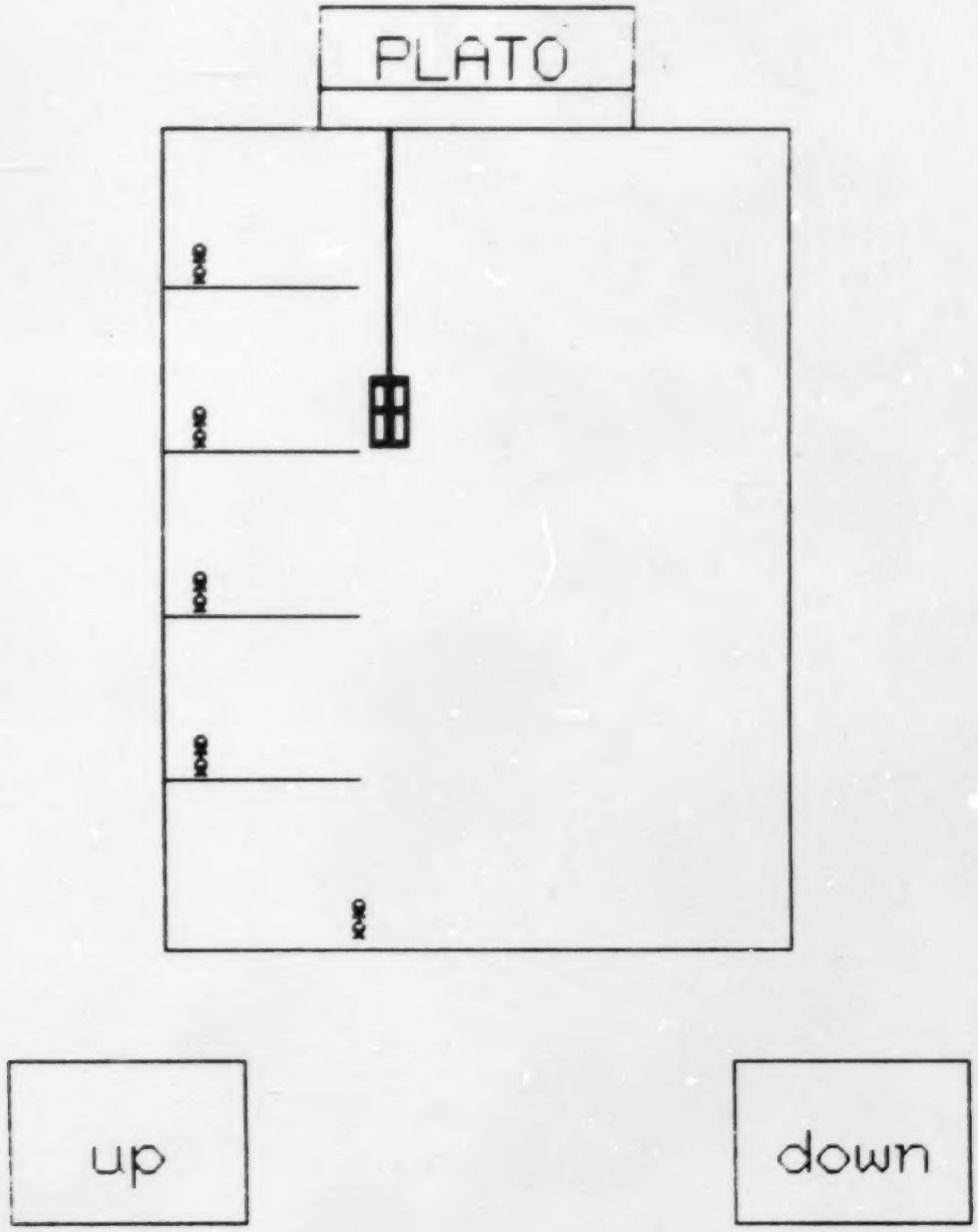


Figure 2. The figure on the bottom floor says, "Send the elevator to me, please." The student has to read the words and understand which button to push.

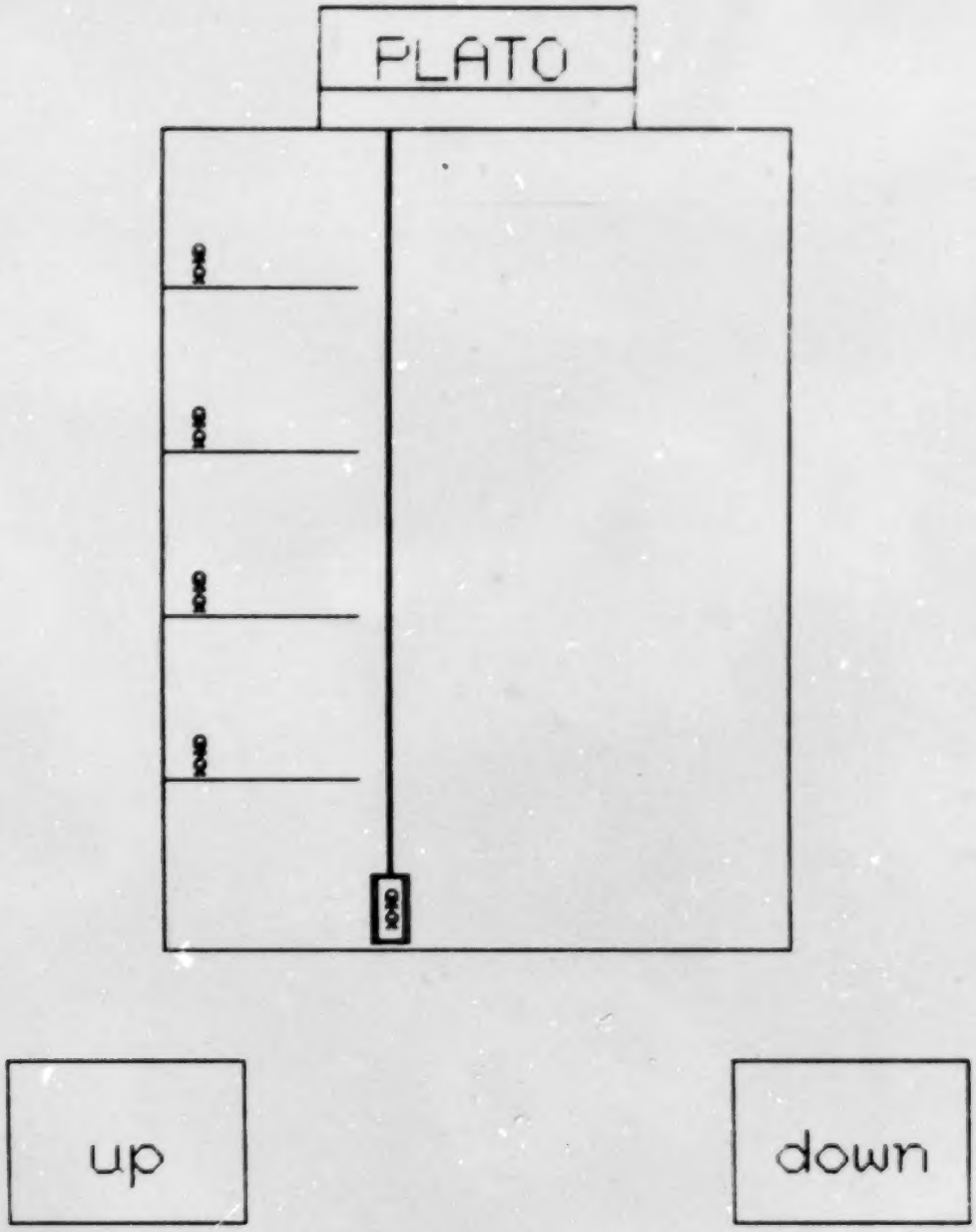
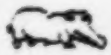


Figure 3. Once the student moves the elevator to the correct floor, the figure hops on the elevator and says, "I want to go up, please."

 alligatorhippo	hippoelephant			

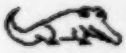
hippo

monkey

zebra

Make an animal!

camel


alligator

kangaroo

Type the names of two animals

> kangaroo-monkey

elephant

seal

lion

octopus

Figure 4. In the "Animal Maker" lesson, students can make up to five animals at a time. So far, this student has "made" an alligator-hippo and a hippo-elephant; now the student wants to see what a kangaroo-monkey looks like.

Figure 5. In "The Race", students decide how fast a dog will run.

The dog will run fast.

Figure 6. In this case, the student wants the dog to run fast. Another feature of this lesson is that the students can touch the little square box in front of the sentence at the bottom to have the audio unit read the sentence to them.

Figure 7. Now the students have to decide how fast the cat will run. In this telling of the story, the cat will run slowly.

Figure 8. Now the students decide whether the bridge should be up or down. In this case, the bridge will be up!

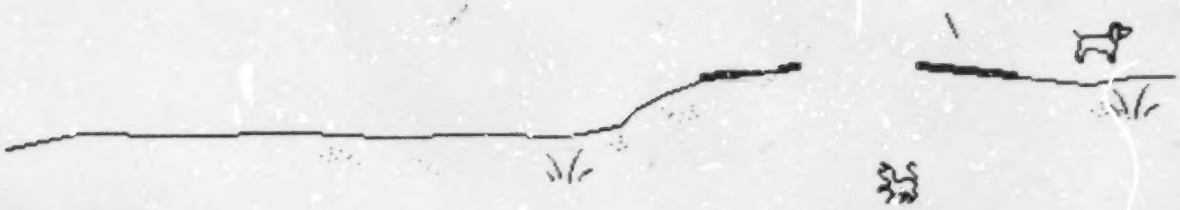


Figure 9. The race goes as predicted. The dog runs so fast that he can jump right over the raised bridge. But the cat is going too slow to make the jump and falls off the bridge.

- The dog is happy.
- The cat is sad.

Figure 10. If the students don't like the way this story turned out, they can do it over again and make different choices.

"Make a Sentence" (figures 11, 12, and 13) is probably the best example of allowing the students to use the computer as a machine. The following sixteen words are written on the screen: the, from, to, over, boy, girl, cat, dog, rabbit, house, tree, car, runs, walks, carries, jumps. The students touch the words they want and try to form them into a sentence; if the sentence is syntactically correct, the computer displays an animation of that sentence. For example, "the rabbit runs" produces a rabbit running across the screen.

The sentences can be quite complex; for example, "the girl walks the cat from the tree to the rabbithouse over the dog." And the sentences can make semantic nonsense and still work; after all, one of the joys of words is that they can symbolize such silly thoughts as, "the tree jumps over the house".

There are several "Make a Sentence" lessons with different vocabularies; the most powerful lesson can accept over 250,000 possible sentences, thus assuring success for even the most imaginative students.

The five lessons described so far demonstrate how entire lesson strategies can be built around the goal of giving the students control over the computer. In all of those examples, the students practice an important reading skill in a context which makes that skill important; students acquire specific reading skills in the process of making the computer "work".

The rabbit runs

to	from	over	the
house	boy	rabbit	dog
tree	girl	cat	car
walks	jumps	carries	runs

ERASE Help available NEXT

Figure 11. In "Make a Sentence", the students have sixteen words with which they can construct a sentence. If they can construct a syntactically correct sentence, the computer animates it. The rabbit above is in the midst of running across the screen.

The girl walks the cat from the tree to the rabbit house over the dog

Figure 12. "Make a Sentence" allows complex sentence such as the one above. A bridge is drawn to graphically represent the concept of "over the dog". Notice, also, that compounds such as "rabbithouse" are acceptable.

The tree jumps over the house

Figure 13. Semantically absurd sentences are also recognized by the "Make a Sentence" lesson. In this picture, the dotted lines shown represent the tree "jumping" over the house.

In many lessons, there is no way to build an entire lesson strategy which puts the students in control of the terminal; for example, drill and practice strategies require a structured approach which does not leave much room for student intiative. Nevertheless, PERC lessons follow lesson design principles which govern the interactions within lessons, and which give the students control over their computer terminals in subtle ways. The following is a list of some of these principles:

(1) All responses should be meaningful; they should be tied as closely to the skill being learned as possible. For example, in the "Hiding Word" lesson (figure 14), students have to uncover a word hiding under some solid blocks and match it to one of two other words on the screen; each solid block covers a letter. The objective of that lesson is to teach students to attend to word detail; and the words are always very similiar. As students touch each solid block, it changes into a letter; thus the response is linked directly to the behavior the students are supposed to acquire.

Well-designed machines do not require humans to go through unnecessary maneuvers in order to operate them. If the computer terminal is simply a machine, then all interactions which do not promote learning should be stripped away from lessons.

(2) Remedial feedback should be kept to a minimum. This flies in the face of the conventional wisdom which celebrates individualized remediation as one of the greatest benefits of CAI. But there are a number of good reasons for minimizing remedial feedback:

(2.1) There is no way to really know WHY students make a given error; therefore, it is useless make differentiated responses to them based on the type of error they made.

tired | tried

Figure 14. In "The Hiding Word" solid blocks hide the letters of the word at the bottom. The students uncover one letter at a time until they can tell which of the words at the top is "hiding". The task of uncovering each letter links the students' response to the skill of attending to the word detail.

(2.2) Students are often positively reinforced by remedial sequences; if their goal has been to make the computer terminal "work", the remedial sequence is proof to them that they have been successful.

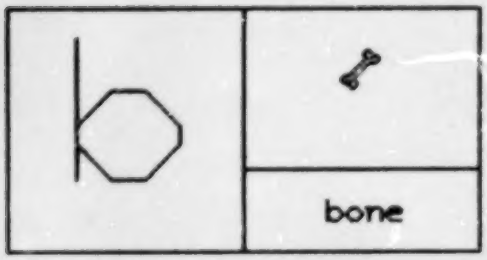
(2.3) Remedial sequences are often more spectacular than the main line instruction; it is difficult to correct students' responses without engaging in a great deal of audio and graphics. Students typically enjoy the remedial sequences more than the main line instruction and, again, feel positively reinforced for having triggered remediation.

Therefore, wrong responses are handled the same way that other machines react when they are incorrectly operated: nothing happens except that the students are encouraged to try a different answer. For example, in the lesson, "How Well Can You Listen?" (figures 15 and 16), students must pick out the picture that starts with a specified sound; if they touch the wrong one, the machine simply says, "No, try again."

Of course, eventually students may need additional help; that help is described in the next point.

(3) Students must always be forced to make the correct response. In the lesson described above, once the students have made several errors (figures 17 and 18), all of the wrong responses are erased from the screen; and the students are forced to touch the one remaining answer before they can continue.

Machines do not work unless they are operated properly. It is very tempting to cycle to the next item as soon as the student makes an error, but that would take away from the machine-like quality of the computer terminal.



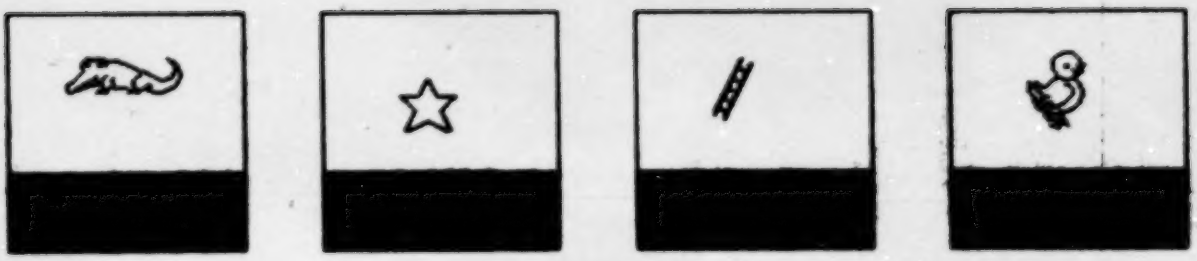
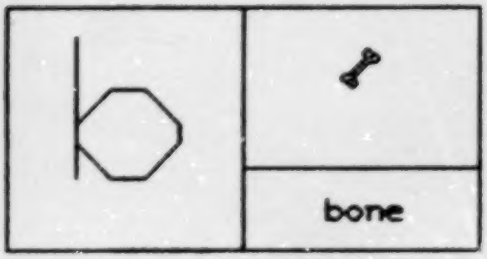


Figure 15. In "How Well Can You Listen", the student must pick out the word that starts with the "b" sound. The letter and an example ("bone") are given at the top; then the audio says each of the four words as the pictures are drawn on the screen. The students can interrupt this sequence at any time if they think they see the correct answer.



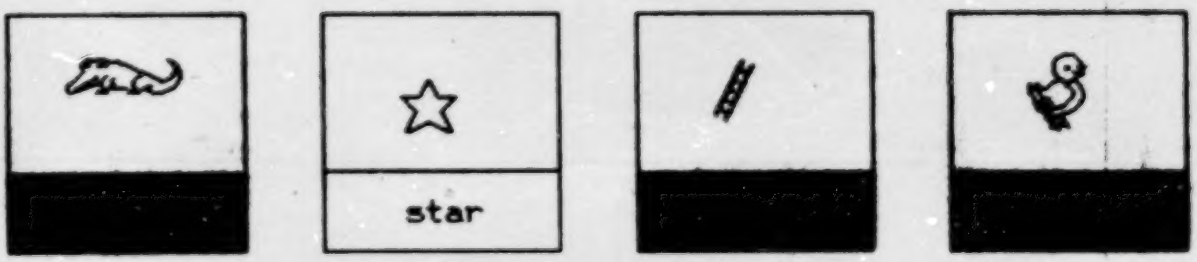
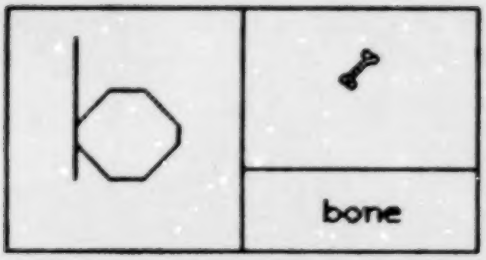


Figure 16. The student has touched the "star", which is a wrong answer. The solid part of the box is uncovered to show the student that the word does not begin with the letter "b"; and the audio tells the student, "No, try again." But there is no additional remediation at this point.



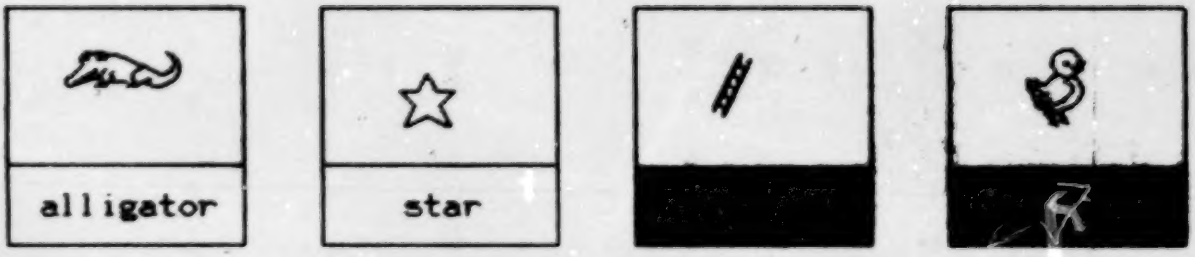


Figure 17. Now the student has touched the alligator, which is his second mistake on this page. First, the solid part of the box is removed to reveal the word "alligator", and to show the student that it does not begin with the letter "b". Then ....

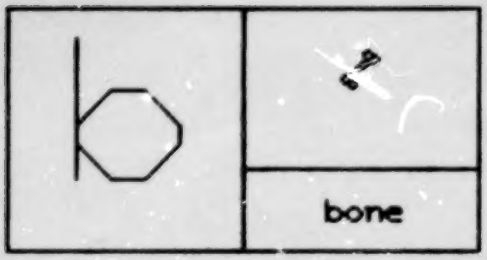


Figure 18. ... all the boxes are erased except the correct answer; the audio says, "No, I'll show you the right one. Touch that one."

(4) Students should be able to pace their own instruction. In "Picture Dictionary" lessons (figures 19 and 20), the students are presented with a letter to touch, such as "m"; each time they touch the "m", they are shown an "m" picture, such as, "m is for moon". The students decide for themselves when they want to proceed; some students speed through the lesson as quickly as possible, but others spend time examining the pictures and text on the screen.

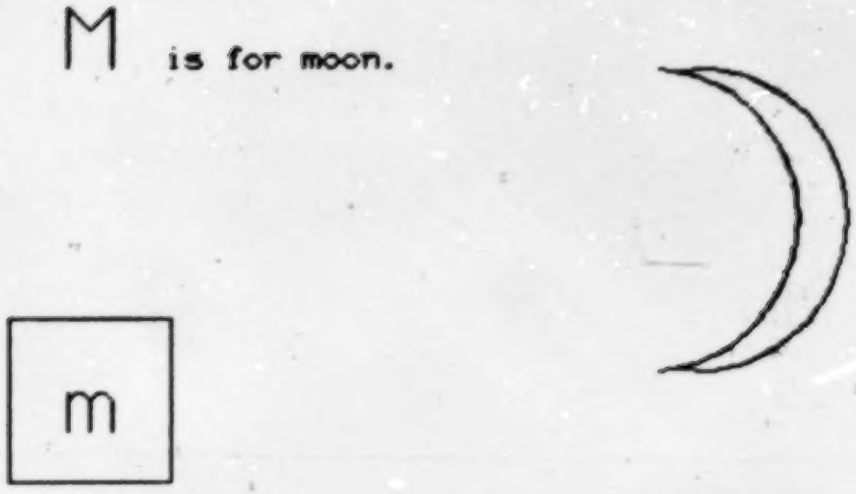


Figure 19. The students start with a page with nothing except an "m" in a box; when they touch the "m" (or type it!), the sentence "M is for moon" is written, the picture of the moon is drawn, and the audio tells them, "M is for moon". Then a new "m" in a box appears and waits for the student to touch it. The students pace themselves through this lesson.

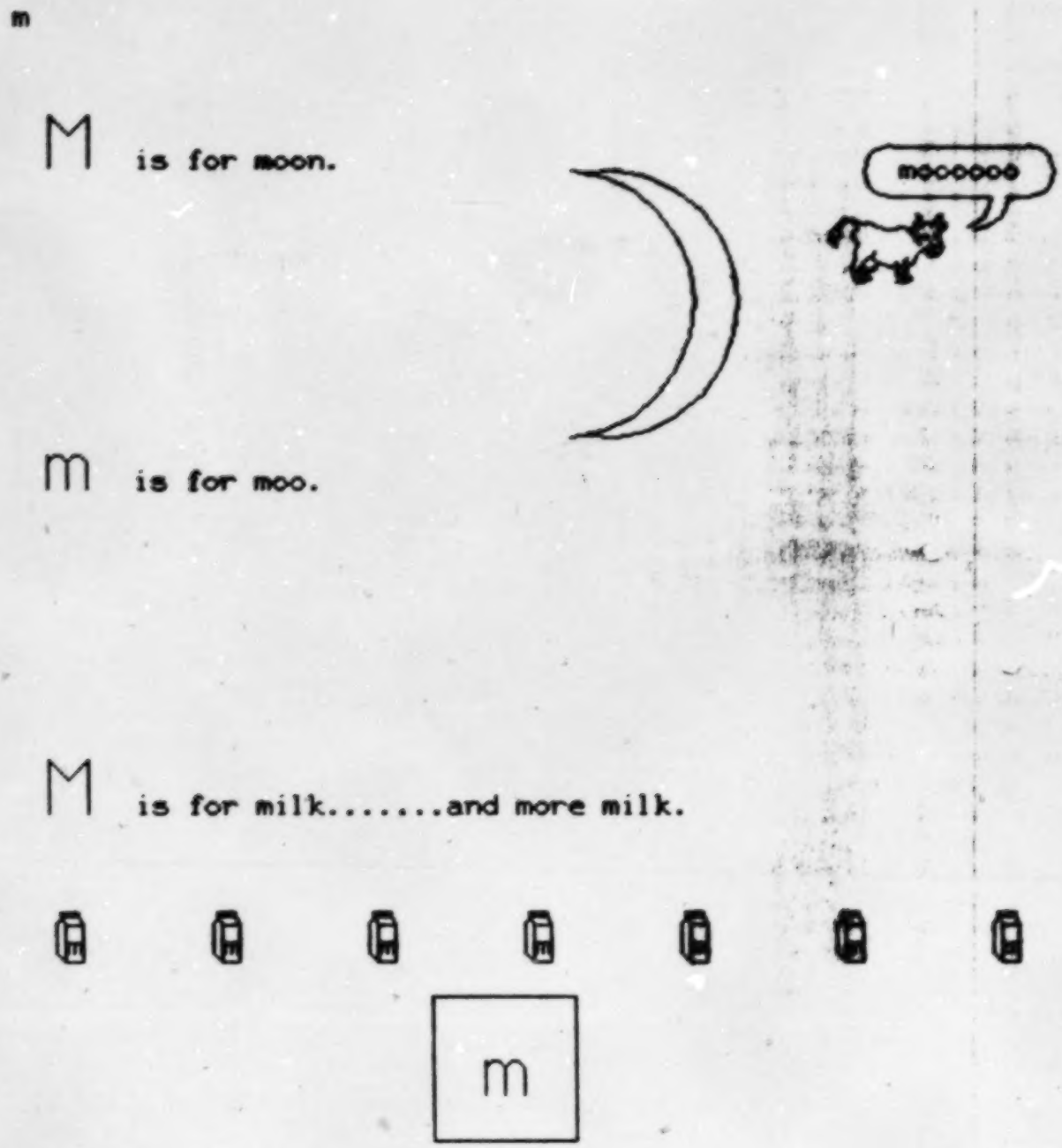


Figure 20. The students build the "Picture Dictionary" displays until they have a full page of "m" words.

The four lesson design principles described above are just some of the conventions which PERC follows. All conventions are subsumed under the one guiding rule which calls for consistency across lessons. Machines are consistent, predictable, and reliable; CAI lessons must be the same. There is nothing more frustrating for students than to have to learn new interaction rules for each new lesson.

A final important characteristic of machines is that they have limits; cars cannot fly, and airplanes would look pretty silly taxiing down our highways. Computer-assisted instruction lessons have limits too. But those limits should not be set too narrowly nor too broadly. Much CAI sets the limits too narrowly; they follow the traditional stimulus-response paradigm with the machine giving the stimulus and the student entering a response. The computer terminal is treated as though it were an extension of the human teacher, a "perfect tutor"; and the students are expected to behave themselves.

Thinking of the computer terminal as a machine means turning the conventional S-R formula around: the student should give the stimulus, and the computer should respond. The student should control the machine.

It is also possible to extend the limits of the machine too far so that the power which the student has over the machine is so great as to be meaningless. One instructional design school advocates "learner control" in which the students are able to choose instructional strategies from moment to moment. But students do not need that much nor that type of control over their lessons; many students do not know what instructional strategies to follow, and they are unlikely to profit from stumbling around trying to find a successful strategy. Furthermore, while educators may not know an awful lot about learning, they can prescribe sequences of instruction which are effective; it does not make sense to make every student rediscover these appropriate instructional strategies in order to learn something.

"Learner control" gives the student too much control. It is like cars with fancy dashboards; they seem to be the latest in technology until it starts to rain and you cannot find the windshield wiper button.

Therefore, the student control which PERC advocates gives students more control than they would have in most other CAI lessons; but not so much that the control becomes meaningless. The computer terminal is viewed as an experiential laboratory; and each lesson is designed to give the students experience with the skills being learned.

Implications

Most CAI research has focused on skill acquisition; researchers have tried to demonstrate that CAI teaches faster, better, or cheaper. While some of this research is undoubtedly necessary, the non-quantifiable effects of CAI have been largely ignored.

The approach to CAI described here puts the emphasis on the process of learning rather than on the content. This implies that some research should be directed at studying the effects of using CAI independently of the content being studied. Based on PERC's experience over the past few years, the following areas would seem to be useful to study:

(a) Locus of control. PERC lessons are designed to make the student feel in control of the computer terminal. Do the students who use PERC lessons develop confidence in themselves sooner; or is there a qualitative difference between students who use PERC lessons daily and those who do not?

(b) Creativity. PERC teachers report that their students are more creative than when the teachers did not have the computer terminals; and that students often draw pictures and develop stories based on ideas which they saw on the computer terminal. PERC lessons encourage students to explore the limits of the machine, and to experiment with words; does such interaction with computer terminals promote creativity in other areas?

(c) Symbolism. In order to operate the computer terminal, students must perform many abstract acts, such as typing a special key to signal the computer that they are ready to proceed. Through such experiences, do students begin to understand the symbolic nature of

reading? If students can understand that typing a special key can represent "ready to proceed", will they better comprehend that a grapheme can represent a phoneme?

(d) Logical abilities. If CAI lessons are programmed so that they are consistent and predictable, do students begin to form expectations and hypotheses about those lessons which promote reasoning abilities? Do students begin to process information in ways which allow them not only to operate the computer terminal, but also to make sense of the world around them?

(e) Effects on learning styles. PERC lessons use a multi-media approach and require the students to interact in many different ways. Will these kinds of experiences broaden the learning abilities of typical students? And can they reach students who normally do not learn very much from traditional educational experiences.

Conclusions

The screen on a PLATO terminal is about eight inches square; all four edges of the screen are always lit up by an orange glow; the borders of the screen must stay lit in order for the screen graphics to work. The orange glow really looks like a halo around the screen. In fact, there is an "orange halo effect" around all of PLATO and most other CAI; students love to work on computer terminals. They are transfixed by the "orange halo".

But it is not always going to be that way. Unless CAI searches out its unique role in reading instruction, it will go down as just another educational fad. This paper has suggested that the perspective from which CAI should be developed is that the computer terminal should be thought of as a machine, and that students should be given control over that machine; and it has described some of the 200 lessons which have been developed by PERC which have tried to implement that approach.

Acknowledgements

I have designed, programmed, or supervised the development of most the lessons described in this paper. But the whole Elementary Reading staff and cooperating teachers have been involved in the evolution of the lessons and ideas described here; Priscilla Corielle, especially, identified some of the lesson design principles early in the life of the project.

Freddy Frog was written by Kathy Lutz; Bonnie Seiler drew the picture of Freddy.

The Elevator was conceived of by Roberta Stock and originally programmed by John Eisenberg.

The Animal Maker was my idea, and originally programmed by Jim Gilfillan.

The Race was written and programmed by me. Priscilla Corielle and John Risken developed the original "branching story" on PLATO ("The Glumph", written by Priscilla, designed and programmed by John); and The Race owes much to The Glumph. John Risken also developed "touch sensitive text" which allows the students to hear any line of a story they wish (mentioned in the description of figure 6).

Make a Sentence was my idea. John Risken and John Eisenberg designed and programmed the original version of the lesson. Tom Schaefges worked on later versions. And I have designed and programmed the current version.

The Hiding Word evolved from an idea programmed by Paul Tenczar (I am not sure if the idea originated with Paul, but the idea for the lesson came directly from Paul's work); the current version is genuinely a creation of the entire reading staff.

How Well Can You Listen was designed and programmed by me.

The Picture Dictionary for "M" was originally written by Grace Yang.

Lee Fillman designed almost every drawing seen here; and she is responsible for the high quality animations in the lessons.

And, finally, every programmer who worked for the project made their own unique, but quiet, improvements to the lessons as the lessons were refined over the years.

Other papers on the Plato Elementary Reading Project

Elementary Reading on PLATO IV, 1977. Urbana, Illinois: Computer-based Educational Research Laboratory, 1977.

Obertino, Priscilla. "The PLATO Reading Project: An Overview." Educational Technology, Vol XIV, No. 2 (February, 1974), pages 13-18.

The PLATO Elementary Reading Curriculum, 1974. Urbana, Illinois: Computer-based Education Research Laboratory, 1974. ERIC number, ED 105 377.

Risken, John and Ed Webber. "A Computer Based Curriculum Management System." Educational Technology, Vol. XIV, No. 9, pages 38-41.

Yeager, Robert. "CMI and the Logical Analysis of Curriculum". Proceedings of the Association for the Development of Computer-Based Instructional Systems (ADCIS). Wilmington, Delaware, February, 1977, pages 55-60.

Yeager, Robert. "Lessons Learned in the PLATO Elementary Reading Curriculum Project". Annual Meeting of the American Educational Research Association, New York, April, 1977. (Currently being indexed in ERIC)

Yeager, Robert. "Using Audio With CAI: Experiences of the Plato Elementary Reading Project." Proceedings of the Association for the Development of Computer-Based Instructional Systems (ADCIS). Minneapolis, Minnesota, August, 1976, pages 227-232.